

Geographic Information and Node Selfish-Based Routing Algorithm for Delay Tolerant Networks

Fang Lu, Jianbo Li*, Shan Jiang, Youmei Song, and Fushu Wang

Abstract: In Delay Tolerant Networks (DTNs), some routing algorithms ignore that most nodes are selfish, i.e., nodes are willing to use their own resources to forward messages to nodes with whom they have a relationship. In view of this phenomenon, we propose a routing algorithm based on Geographic Information and Node Selfishness (GINS). To choose a forwarding node, GINS combines nodes' willingness to forward and their geographic information to maximize the possibility of contacting the destination. GINS formulates the message forwarding process as a 0-1 Knapsack Problem with Assignment Restrictions to satisfy node demands for selfishness. Extensive simulations were conducted, and results show that GINS can achieve a high delivery ratio and a lower hop count compared with GRONE and LPHU. Furthermore, its overhead ratio is 25% and 30% less than that of GRONE and LPHU, respectively.

Key words: delay tolerant networks; node willingness; routing algorithm; geographic information; forwarding process

1 Introduction

Delay Tolerant Networks (DTNs)^[1] are an emerging network architecture that originates from the interplanetary Internet^[2] and exhibits network topology partition, node mobility, and extremely long delivery latency. DTNs have become a hot research topic and a significant challenge in the field of wireless network. As a result of these characteristics, a complete path does not exist between the source and the destination. Consequently, obtaining significant achievements is difficult for traditional routing algorithms based on TCP/IP. To deal with this problem and provide communication services, DTN routing adopts the store-carry and forward mechanism to spread messages hop by hop. However, because of the randomness of intermittent connectivity between nodes, obtaining global network topology knowledge and information

about the destination may be unrealistic. Therefore the key issue is the selection of the optimal next hop relay nodes to increase message delivery ratio and decrease resource consumption and network overhead.

Numerous routing algorithms have been proposed to address this difficult routing problem, such as epidemic^[3], which is a typical flooding-based algorithm, and MaxProp^[4], which can reduce the overhead of epidemic routing. Furthermore, some routing algorithms aim to achieve optimal system performance^[5]. Although these routing algorithms can increase message delivery reliability, they do not consider the willingness of nodes and implicitly assume that all nodes are willing to forward messages to other nodes.

In real scenarios, some selfish nodes may not be willing to forward messages to others, such as in mobile social networks^[6,7] and PeopleNet^[8]. The previously mentioned routing algorithms may not work well in such a network environment, because some messages are forwarded to nodes that are unwilling to relay and will be dropped. To capture node selfishness in a more realistic manner, we have three observations from the social perspective. First, selfish nodes are usually willing to help others close to themselves (e.g., relatives, friends). Second, the selfishness of a node

• Fang Lu, Jianbo Li, Shan Jiang, Youmei Song, and Fushu Wang are with the College of Computer Science and Technology, Qingdao University, Qingdao 266071, China. E-mail: 1105627228@qq.com; lijianbo@188.com.

* To whom correspondence should be addressed.

Manuscript received: 2017-03-16; revised: 2017-03-23; accepted: 2017-03-28

is reflected in the pursuit of the highest forwarding profit, that is, a node is willing to deliver more important messages to those nodes with whom it has social ties, especially when resource constraints exist. Third, nodes show different degrees of selfishness for different messages: a higher importance of the message corresponds to a lower degree of selfishness shown by the node. Currently, most existing node selfishness-based routing algorithms either take the first two observations^[9,10] into consideration or take only the last observation^[11] into consideration. An effective selfish routing strategy that comprehensively considers these three observations is still lacking. A DTN routing algorithm should take various situations of node selfishness into consideration.

To address the abovementioned challenges, we propose a geographic information and node selfishness based routing algorithm for DTNs. To select a good next hop relay node, we combine the relay node's willingness and geographic information. The relay node's willingness is used to evaluate its forwarding capability and thus reduce the message dropping rate. We use the relay node's geographic information to spread the message to the direction of the destination as far as possible, thereby increasing delivery ratio and reducing network load. To maintain node selfishness, Geographic Information and Node Selfishness (GINS) allocates resources such as buffers based on message priority which is related to the relationship among nodes. Furthermore, for the purpose of pursuing maximum selfish profit among nodes, GINS formulates the forwarding process as a 0-1 knapsack problem with assignment restrictions. Our major contributions are listed as follows:

- An evaluation criterion called forwarding willingness degree is proposed for evaluating forwarding willingness for other nodes, and we use it as the next hop node's limiting condition.
- A geographic routing scheme based on neighbor node information is proposed to select the next nodes. A node needs only its local neighbors' geographic position information and forwarding willingness to make the next hop choice among neighbor nodes with whom it has social ties.
- A mechanism is proposed to establish a buffer and message management strategy based on priority. The message forwarding set is based on message priority and combines with node buffer size, thereby improving the algorithm message delivery ratio.

- 0-1 knapsack problem with assignment restrictions is used to formulate the message forwarding set process and provide a heuristic solution.

The rest of this paper is organized as follows: In Section 2, we discuss some related works. In Section 3, we introduce the preliminary and motivation of the routing strategy. In Section 4, we describe the GINS scheme in detail. Simulation evaluations and analysis of results are presented in Section 5. In Section 6, we conclude our paper.

2 Related Work

Recent years have seen considerable research that deals with the issues of routing algorithms in DTNs. Most of such research is devoted to enhancing connectivity on demand. Some algorithms introduce additional communication facilities into the network, such as message ferry, and a few assume a predictable mobility pattern or employ various information distribution strategies to diffuse the needed knowledge about network topology. However, these algorithms may increase the difficulty of executing these protocols in reality.

To reduce the computational complexity and control congestion in the network, a set of congestion control schemes for DTNs have been proposed. In Ref. [12], the bridge protection algorithm, which changes the behavior of a set of topologically important nodes in the network, was proposed. In Ref. [13], two different schemes were presented, which maximize the delivery ratio and minimize the overhead ratio. In Ref. [14], a geographic routing protocol was proposed for MANETs in delay tolerant situations, by using no more than one-hop information. However, these algorithms ignore social attributes such as the selfishness of most nodes, which means that they will result in a high message drop rate, thereby affecting the performance of the routing algorithm. Other research achievements have been made for transmission in DTNs. In Ref. [15], DTN routing was treated as a resource allocation problem that translates the routing metric into per-message utilities that determine how messages should be replicated in the system. In Ref. [16], a utility function was proposed as the difference between the expected reward and the energy cost spent by the relay to sustain forwarding operations. A user selfishness-based approach for calculating the optimal policy was developed in Ref. [17]. Although Ref. [17] takes the node selfishness into account, it ignores the possibility

that nodes with weak ties might be a prospective future good because the node might move to a farther place and encounter more different nodes. Our work mainly differs from previous research in three aspects. First, we use **statistical analysis** to understand relations between nodes accurately. Second, we implement **a relay node selection strategy** based only on **the node selfishness and the available position information** of neighbor nodes. Third, we introduce a new approach to allocate network resources reasonably. Simulation results show that our algorithm has a high delivery ratio and the lowest hop count on average, as well as an acceptable overhead ratio.

3 Preliminary and Motivation

In DTNs, some routing algorithms make the next hop decision by obtaining more than one-hop neighbor information and even requiring global network information, such as well-known distributed routing algorithms. The partitioned network topology and high end-to-end delay may result in out-of-date collected information even though information was obtained in real time. Moreover, frequent disruptions in DTNs may cause messages to flood the entire network and thus result in high network loads. Other problems in existing routing algorithms also require attention^[13,14]. For instance, routing algorithms ignore the selfishness of nodes in the real world, thereby resulting in a high message drop rate.

To overcome the disadvantage of current routing schemes, we propose a new routing scheme that relies on node selfishness and the position information of its one-hop neighbor for each node. Our primary objective is to **maximize the delivery ratio**. In addition, **the average hop count and the overhead ratio** should be **controlled** to an acceptable level. High delivery ratio may benefit from quick delivery in DTNs, and the assumption is that shortening the distance between the current node and the destination node can increase the message delivery ratio. Based on this concept, we design a routing scheme to spread the message to the direction of the destination as far as possible. By referring to the abovementioned factors, we refine the most important principles of designing our routing algorithm as follows:

(1) No assumption of available global knowledge is made, and the node can obtain its position by related positioning algorithm and it can broadcast its location to its neighbor nodes.

(2) The node collects the encounter history of node pairs in every meeting opportunity.

(3) In the entire network, all nodes are selfish, and the number of message copies is controlled by some methods to avoid broadcast storming.

4 Routing Details

In this section, we first present a detailed introduction of node selfishness. Then, we describe how relay nodes are selected, as well as present a discussion and analysis of message and buffer management. Finally, we present our overall routing process.

4.1 Node selfishness

Selfishness is not a special phenomenon of individual nodes, but it is a common phenomenon in real life. Therefore, our paper assumes that all nodes in delay tolerant networks are selfish. However, under different circumstances, the degree of selfishness of nodes is exhibited in two aspects. **The first aspect is whether the node is willing to forward messages for neighbor nodes;** we refer to this behavior as the forwarding willingness degree. **The second aspect is that the selfish behavior of nodes tends to achieve the most selfish profit.**

4.1.1 Forwarding willingness degree

The forwarding willingness degree is determined by the relationship between nodes. In this paper, by collecting the encounter count information between two nodes, we use **statistical analysis** to define **a utility value** as a forwarding willingness degree metric, which can be used to estimate whether a node is willing to provide better service for neighbor nodes. In other words, the forwarding willingness degree metric is used to evaluate the forwarding capability of nodes. For this purpose, a DTN node needs to maintain an $N \times N$ dynamic matrix **DM** (i.e., Eq. (1)) to record the encounter information with other N nodes in the total time. The $C_i(\text{node}_j)$ denotes the encounter counts between node_{*i*} and node_{*j*} in the total time unit. Then, all encounter history information is stored in the **DM**.

$$\mathbf{DM} = \begin{bmatrix} C_i(\text{node}_1) & \cdots & C_i(\text{node}_N) \\ \vdots & \ddots & \vdots \\ C_N(\text{node}_1) & \cdots & C_N(\text{node}_N) \end{bmatrix} \quad (1)$$

With the **DM**, we define several statistics as follows:

(1) Average (AVG)

The $P_i(j)$ shown in Eq. (2) represents the probability that $C_i(\text{node}_j)$ may occur. The AVG, which is the

average level of all samples that can reflect the central tendency, is defined in Eq. (3).

$$P_i(j) = \frac{1}{N} \quad (2)$$

$$AVG_i = \sum_{k=1}^N C_i(k) \times P_i(k) = \frac{\sum_{k=1}^N C_i(k)}{N} \quad (3)$$

(2) Variance (VAR)

The VAR defined in Eq. (4) can comprehensively reflect the degree of dispersion of all variables, which is often used to describe the stability of data, and is defined in Eq. (4).

$$VAR_i = \sum_{k=1}^N (C_i(k) - AVG_i)^2 \times P_i(i) = \frac{\sum_{k=1}^N (C_i(k) - AVG_i)^2}{N} \quad (4)$$

(3) Covariance (COV)

The COV defined in Eq. (5) is typically used to analyze the relationship between two variables, which describes the relationship between the changes of the two variables.

$$COV(i, j) = \frac{\sum_{k=1}^N (C_i(k) - AVG_i)(C_j(k) - AVG_j)}{N} \quad (5)$$

Finally, with the above statistics, we further normalize the COV by the production of multiple VAR, as in Eq. (6). We can objectively evaluate the relationship between a neighbor node and the current node by using Eq. (6), where $F_D(i, j)$ denotes the forwarding willingness degree. The value of $F_D(i, j)$ is a number within $[-1, 1]$, when $F_D(i, j) > 0$, which indicates that social ties exist between node_{*i*} and node_{*j*}, and node_{*j*} is willing to help node_{*i*} to forward messages. Otherwise, node_{*j*} is unwilling to forward messages for node_{*i*} with whom it has no social relationship.

$$F_D(i, j) = \frac{COV(i, j)}{\sqrt{VAR_i} \sqrt{VAR_j}} = \frac{\sum_{k=1}^N (C_i(k) - AVG_i)(C_j(k) - AVG_j)}{\sqrt{\sum_{k=1}^N (C_i(k) - AVG_i)^2} \sqrt{\sum_{k=1}^N (C_j(k) - AVG_j)^2}} \quad (6)$$

4.1.2 Message forwarding profit

In DTNs, node selfishness is also reflected in the pursuit of the highest message forwarding profit. The size of the message forwarding profit is related to the priority of the message and the relationship between nodes.

Thus, when a social relationship exists between nodes, a high message priority corresponds to a large message forwarding profit. In our paper, the node preferentially sends a message with the highest forwarding profit to a node with whom it has a social relationship. Therefore, we take the number of nodes with the message and survival time of the message as the comprehensive criteria for measuring priority and make message a higher priority, which has a wider distribution and is more urgent. Equation (7) can calculates the final priority of each message.

$$UP_m = \frac{N(m)}{N} \times \frac{TTL}{TTL - L(m)} \quad (7)$$

where $N(m)$ denotes the number of node with the message m , N denotes the total number of nodes in the entire network, TTL denotes the living time of message m , and $L(m)$ denotes the time that message m has survived.

Definition 1 (Message forwarding profit) A high priority of the forwarded message and a great success rate of the forwarded message correspond to a large forwarding profit. Therefore, message forwarding profit is defined as the product of message priority and the increment of the successful delivery rate of the message. The calculation formula is as follows:

$$G_m = UP_m \times I_p \quad (8)$$

where UP_m denotes the priority of message m , $I_p = 1 - (1 - P_{(j, d_m)})(1 - P_{(i, d_m)}) - P_{(j, d_m)}$ denotes the successful delivery probability increment of message m , whose message m from node_{*j*} is forwarded to node_{*i*}. $P_{(i, d_m)}$ and $P_{(j, d_m)}$ represent the probability that node_{*i*} and node_{*j*} will encounter the destination. The calculation formula is as follows:

$$P_{(i, d_m)} = \frac{C_i(\text{node}_{d_m}) + \lambda}{\sum_{k \in S_i} C_i(\text{node}_{e_k}) + N\lambda} \quad (9)$$

where $\sum_{k \in S_i} C_i(k)$ denotes the total encounter count of node_{*i*} and all neighbor nodes for the entire time. λ is a prior tail number in the formula. Equation (9) is derived from the theorem below. We introduce the theorem in advance. Then, the corresponding reasoning process is given.

Theorem 1 Beta distribution. The unknown parameter is a random variable. A density function of the beta formula is denoted in the following:

$$P(\theta_i|x) = \frac{P(x|\theta_i)P(\theta_i)}{\sum_{i=1}^K P(x|\theta_i)P(\theta_i)} \quad (10)$$

Among $P(x|\theta_i)P(\theta_i)$ indicates the full probability formula, and x is also regarded as a random variable.

bayesian公式

F_D作为下一跳的选择限制条件

tail number ?

Theorem 2 Dirichlet distribution. Dirichlet distribution is a multinomial distribution. μ is the parameter of multinomial distribution. The Dirichlet distribution function form is as follows:

$$\text{Dir}(\mu|\alpha) = \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_K)} \prod_{k=1}^K \mu_k^{\alpha_k-1} \quad (11)$$

where $\alpha_0 = \sum_{k=1}^K \alpha_k$, $\alpha = (\alpha_1, \dots, \alpha_K)^T$ is the parameter of Dirichlet distribution.

Proof Equation 9 combines knowledge of Dirichlet distribution and Beta distribution. C is a collection of the meeting counts between the current node and the other nodes. We assume that P is subjected to Dirichlet distribution, $P \sim \text{Dir}(\delta)$. $P(P_1, \dots, P_N) \sim \delta(\delta_1, \dots, \delta_N)$.

$$\begin{aligned} P(P|\delta, C) &= \frac{P(P, \delta, C)}{P(\delta, C)} = \\ &= \frac{P(C|P, \delta)P(P|\delta)P(\delta)}{P(\delta) \int P(C|P, \delta)P(P|\delta)dP} = \\ &= \frac{P(C|P, \delta)P(P|\delta)}{\int P(C|P, \delta)P(P|\delta)dP} \frac{\prod_{i=1}^N P(C_i|P, \delta)P(P|\delta)}{\prod_{i=1}^N P(C_i|P, \delta)P(P|\delta)} = \\ &= \frac{\prod_{i=1}^N P(C_i|P, \delta) \frac{1}{\Delta(\delta)} \prod_{i=1}^N P_i^{\delta_i-1}}{\prod_{i=1}^N P_i^{C_i+\delta_i-1} \frac{Z}{Z'}} = \\ &= \frac{Z}{Z'} (Z' = Z \Delta(\delta)) \end{aligned}$$

We can draw $P \sim \text{Dir}(C + \delta)$, $E[P_i] = \frac{C_i + \delta_i}{\sum_{i=1}^N C_i + \delta_i}$, $P_i = \frac{C_i + \delta_i}{\sum_{i=1}^N C_i + \delta_i}$. δ_i is a prior tail number. To derive $P(P|\delta, C)$, we need to perform N multiplication operations and 1 division operations. Thus, we obtain $\text{Cost} = N + 1 = O(N)$. With the cost controlled to $O(N)$ level we can finally compute $\frac{\prod_{i=1}^N P_i^{C_i+\delta_i-1}}{Z'}$, and then obtain $E[P_i] = \frac{C_i + \delta_i}{\sum_{i=1}^N C_i + \delta_i}$.

4.2 Utility function for choosing relay node

When network resources such as buffer capacity are sufficient, using more relay nodes typically means a higher delivery ratio. However, if message copies are distributed among several nodes that are unwilling to forward, then these messages will most likely to be dropped. Moreover, using more relay nodes also means that message redundancy may occur. Nevertheless, network resources in DTNs are usually strictly

constrained. Redundant messages use a considerable amount of network resources and easily cause network congestion. Consequently, routing in DTNs should take all the above mentioned factors into account. We have no information about the destination node, e.g., the distance or the direction. Thus, the only way to maximize the possibility of contacting the destination is to spread messages as uniformly as possible.

Thus, we combine a node's forwarding willingness degree and geographic information to choose the relay node, that is, our method allows nodes to choose their own relay nodes that are willing to forward in each semi-circle area opposite the current node. As shown in Fig. 1, node S is the source node of a certain message, and the circle represents the communication range. Node S is the node that is currently chosen to spread the message farther, and we create a line vertical to the line SD and hence obtain two parts that contain nodes A and B in node D 's semi-circle area. If the routing can continue to operate in this manner, then the message coverage area would increase and finally cover the destination.

When the direction is determined, choosing a farther relay node may reduce the superfluous message coverage area, as illustrated in Fig. 2. The area of the intersection of solid and dotted lines denotes the superfluous message coverage area. The superfluous message coverage area of the farthest node is far smaller

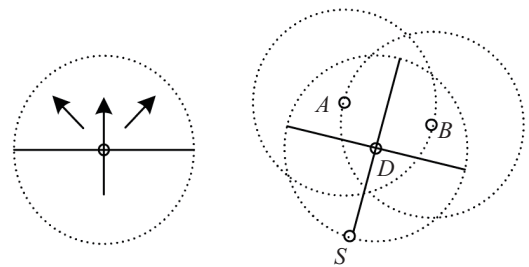


Fig. 1 Selection of relay nodes according to radiation pattern.

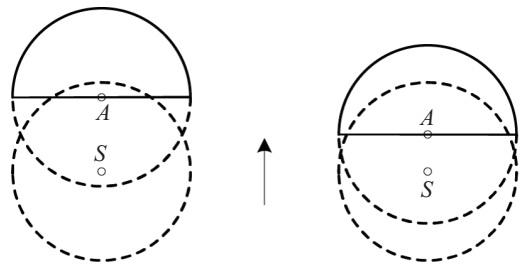


Fig. 2 Superfluous message area of two kinds of nodes.

than that of the nearest node. Thus, our task is to obtain the most likely direction of each part and let each node choose the node that is closest to the direction. We know from statistical significance information that an expectation of the direction is the best choice and leads to a minimum variance. And consequently we aim to obtain the expected shown up direction for a node in the section. Because each node has the same probability that shows up anywhere in the area. Thus, node location position information that shows up obeys a uniform distribution, and we obtain the probability density function, as shown in Eq. (12).

$$f_{(i,j)} = \frac{1}{SC_{\text{part}}} = \frac{1}{\frac{1}{4}\pi R^2} \quad (12)$$

We can obtain the expected direction by using Eq. (13).

$$E[\theta] = \iint f_{(i,j)} \theta dSC = \int_0^{\frac{\pi}{2}} \int_0^R \theta \frac{1}{\frac{1}{4}\pi r^2} r dr d\theta = \frac{1}{2} \theta^2 \Big|_0^{\frac{\pi}{2}} \cdot \frac{1}{\frac{1}{4}\pi R^2} \cdot \frac{1}{2} r^2 \Big|_0^R = \frac{\pi}{4} \quad (13)$$

Therefore, we obtain the expected direction, as calculated above, $\frac{\pi}{4}$. We combine the expected direction and distance to define the utility function to choose the next-hop node by using Eq. (14), and $D_{(i,j)}$ and $\theta_{(i,j)}$ denote the distance and the angle between node_{*i*} and node_{*j*}, respectively.

$$U_{(i,j)} = 1 - |R - D_{(i,j)}| / \left(\frac{1}{2R} \right) - \left| \theta_{(i,j)} - \frac{\pi}{4} \right| / \frac{\pi}{2} \quad (14)$$

4.3 Strategy for message and buffer

In DTNs, because of the limited buffer space of nodes, the node may not be able to accept all messages that other nodes send. In this section, we mainly solve two key related problems, namely, how to manage the buffer resource and which messages should be sent.

4.3.1 Buffer management

We manage buffers based on message priority: (1) messages with priority 0 will be dropped; and (2) buffers adopt the priority preemption strategy, that is, when buffer space is insufficient, low-priority messages are dropped first, and a new incoming message can preempt the buffer occupied by low-priority messages. For example, message *m* is a message in node_{*i*} buffer. When node_{*i*} selects node_{*j*} as the next hop node,

message *m* is not only stored in the free buffer of node_{*j*}, but also seizes the buffer of the smaller priority message for storage. Thus, the buffer size that can store message *m* in node_{*j*} is

$$W_m = W_0 + \sum_{\{k | UP_k < UP_m\}} w_k \quad (15)$$

where W_0 denotes the free buffer size of node_{*j*}, w_k denotes the size of message *k*, and $\{k | UP_k < UP_m\}$ indicates the message set that the message priority in node_{*j*} is less than that of message *m*.

4.3.2 Message forwarding set

When the node_{*i*} selects the message forwarding set *M*, it is designed to maximize the forwarding profit according to Theorem 3.

Theorem 3 The message forwarding set can form into an optimal decision problem, which can be further classified as 0-1 knapsack problem.

Proof If the M_k is regarded as a knapsack, then G_k and w_k are regarded as the value and the weight of message *k*, respectively. Thus, the message forwarding set is equivalent to the corresponding 0-1 knapsack problem, thereby proving that the message forwarding set is a 0-1 knapsack problem. Thus, suppose all the messages in *M* are stored by priority in the decreasing order, then we can use *k* to denote the *k*-th message. Let X_k denote if message *k* is selected by the transmitted subset ($X_k = 1$) or not ($X_k = 0$). Based on the above principles, the problem can be established by the following mathematical models according to the 0-1 knapsack problem:

$$\begin{aligned} \max \quad & \sum_{k \in M} G_m X_k, \\ \text{s.t.} \quad & \forall k \sum_{j \leq k} X_j w_j \leq W_k \end{aligned} \quad (16)$$

Thus, we obtain a greedy algorithm, which ranks the messages in decreasing order of message priority and sends them one by one until no more messages can be stored. The details are shown in Algorithm 1. The time complexity of this algorithm is $O(|M|^2)$, which is acceptable because most equipment has such computing capability.

4.4 GINS

With the combination of the above schemes, we finally implement GINS. Figure 3 illustrates how GINS works, and the four steps of the operation of GINS are detailed below:

Algorithm 1 Greedy Algorithm for 0-1 knapsack problem**Input:**

The selfish profit G_m for any message in M sorts M in the decreasing order of message priority;

node_{*i*} selects node_{*j*} as the next hop node;

```

1: Double TotalM = 0;
2: for message k from 1 to |M| do
3:   if  $W_k > w_k$  then
4:     if TotalM +  $G_k \geq$  TotalM then
5:       TotalM +=  $G_k$ ;
6:       Forward message k to nodej;
7:       for message j from k + 1 to |M| do
8:          $W_j -= w_k$ ;
9:   else
10:    continue;
11: return

```

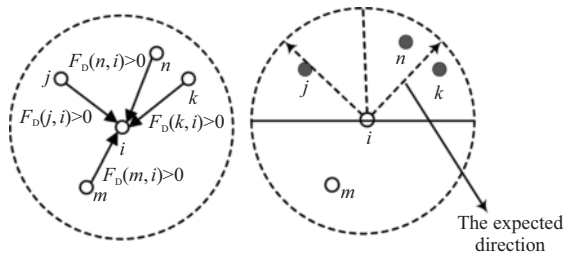


Fig. 3 Selection of relay nodes.

(1) After neighbor discovery, the neighbor node set of node_{*i*} is N_i at the t moment. That is, node_{*j*}, node_{*n*}, node_{*k*}, and node_{*m*} are the neighbor nodes of node_{*i*} in Fig. 3. Nodes deliver messages and perform buffer management in decreasing order of priority.

(2) We choose the next hop node according to the nodes forwarding willingness degree and geographic information. Then, when only one node is willing to forward, the node_{*i*} sends the node a summarized list of messages. When more than one node exists in Fig. 3, node_{*i*} will use the utility function-based method to choose the next hop nodes (Section 4.2).

(3) From the priority information, node_{*i*} calculates the new priority value for each message (Eq. (7)). Based on the new priority and other information, node_{*i*} calculates its delivery probability (Eq. (9)) and the available buffer size (Eq. (15)) for each message.

(4) In accordance with Eq. (8), node_{*i*} calculates the forwarding profit of each message and the message forwarding set M that is suitable for forwarding to the next hop nodes according to decreasing order. Considering the available buffer size information of the next hop nodes, node_{*i*} further decides which messages to transmit by solving the 0-1 knapsack problem.

5 Simulation

5.1 Simulation environment and settings

The results are evaluated by using the ONE^[18] simulator. To conduct simulations, we use 126 DTN nodes in a Helsinki City model-based mobility scenario, which consists of 4500 m × 3400 m area. These nodes are divided into six groups. Groups 1 and 3 are pedestrian groups with speeds of 0.5–1.5 m/s and each group consists of 40 nodes. Group 2 is the car group with speeds of 2.7–13.9 m/s and consists of 40 nodes. Groups 4–6 are tram groups with speeds of 7–10 m/s, and each group consists of two nodes. The simulation uses two devices: a Bluetooth device with a transmission speed of 250 Kbit/s and a transmission range of 20 m, and a high speed device with a transmission speed of 10 Mbit/s and a transmission range of 1000 m. The fourth group uses both devices, whereas the other groups use only the Bluetooth device. The other simulation settings are shown in Table 1.

Finally, we evaluate GINS against other two popular algorithms for performance comparison, i.e., LPHU^[13] and GRONE^[14], in terms of delivery ratio, overhead, and average hop count. We investigate the variations of these metrics based on the buffer size, message generation time, living time of messages, and size of messages. The two comparison routing algorithms are listed as follows:

LPHU: Different policies are implemented on the message source node and the relay node, which combines nodes' local positions and historical utility information.

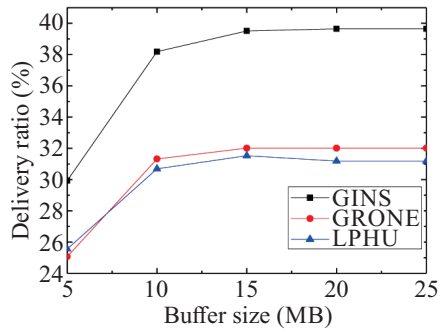
GRONE: This geographic routing algorithm uses no more than one-hop information and considers direction and distance. It also employ a criterion to evaluate the degree of message redundancy.

5.2 Simulation result and analysis

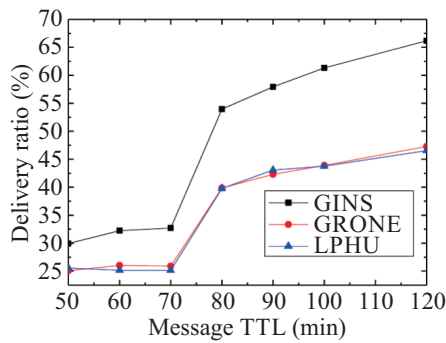
Figure 4 shows the delivery ratio of the three routing protocols. Figure 4a shows that GINS has the highest

Table 1 Settings in the Helsinki City model.

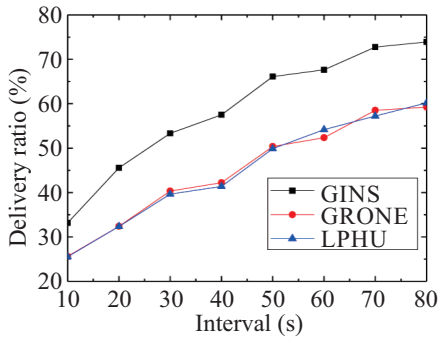
Parameter	Range
Initial topology	Uniform
Message size	300–700 KB
Message interval	10–80 s
Node buffer size	5–25 MB
Simulation time	12 h



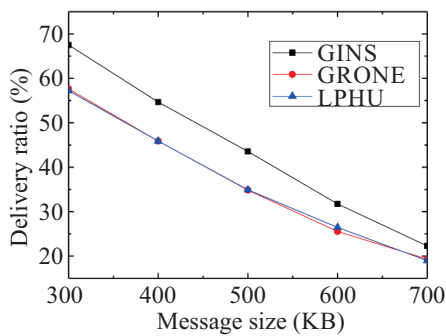
(a) Varying buffer



(b) Varying TTL



(c) Varying interval



(d) Varying size

Fig. 4 Delivery ratio versus buffer size, message TTL, message interval, and message size.

delivery ratio compared to other two protocols because it employs a node's forwarding willingness and utility-based strategy to choose the next hop node instead of choosing blindly. Furthermore, GINS has a message and buffer management mechanism, thus saving the limited buffer resource. In Fig. 4b, as the TTL increases, all algorithms can carry more messages to the destinations. We also observe that the delivery ratio of GINS is the highest among all algorithms, outperforming GRONE, and LPHU by 35%–45% because GINS incorporates several factors such as node forwarding willingness, and geographic information into relay selection. It prevents messages from being forwarded to low-willingness nodes. By contrast, the other two algorithms do not take these factors into account, thereby resulting in many message drops. As shown in Fig. 4c, with the increment of message interval, the delivery ratio of GINS, GRONE, and LPHU schemes increases. The three curves of GINS, GRONE, and LPHU indicate that GINS outperforms LPHU and GRONE by approximately 20%. Figure 4d shows that the delivery ratio of GINS is always greater than that of GRONE and LPHU, thereby indicating that the buffer utilization efficiency of GINS is more efficient.

In Fig. 5, we evaluate the overhead ratio performance of the three algorithms. GRONE implements a replication mechanism based on utility function and message redundancy mechanism that in some sense reduces unnecessary forwarding operations. LPHU is a local position and history utility-based routing algorithm, which limits the replication abilities of messages by relay nodes. These two algorithms do not take selfish nodes into account. When a message is forwarded to a node that is unwilling to forward, it will most likely be dropped. Thus, GRONE and LPHU have a higher overhead ratio than GINS, as shown in Fig. 5. Figure 5a shows that increasing buffer size can be conducive to copying with a high overhead ratio because the delivery ratio would increase, as shown in Fig. 4a, thus increasing the number of delivered messages. Figure 5b shows that GINS has the lowest overhead ratio among all algorithms because it uses more effective message management. Figure 5c shows that the increasing message interval inevitably creates a large number of redundant messages in the entire network, thereby increasing the overhead ratio of the three algorithms. Figure 5d shows that the overhead

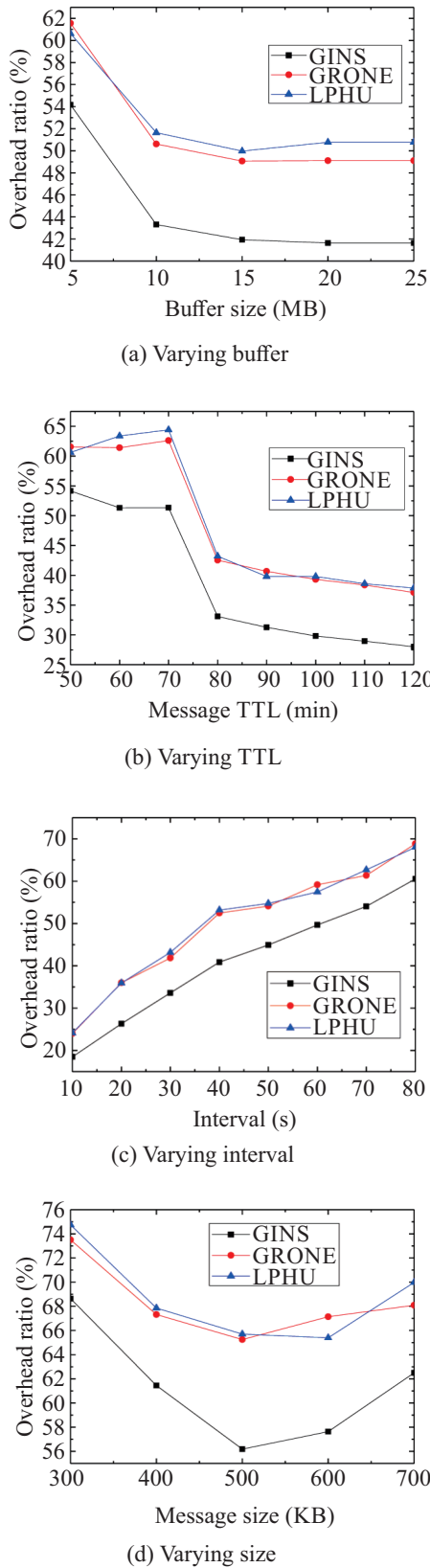


Fig. 5 Overhead ratio versus buffer size, message TTL, message interval, and message size.

ratio of all the algorithms increases with a message size greater than 500 KB because the buffer resource is always limited, thus increasing the message size will increase the message loss.

Figure 6 illustrates the average hop count performance for the three routing algorithms. The results indicate that GINS has the least average hop count among the three algorithms because it makes rational next hop choices, thus causing the message to move toward the destination. As illustrated in Figs. 6a and 6b, buffer size and message TTL do not have a significant effect on the average hop count for all the three routing algorithms. We also observe that the three algorithms maintain an approximately constant average hop count in the case of sufficient buffer resource and message TTL.

Figure 6c shows that the average hop count of GINS, GRONE, and LPHU increases in the wake of the enlargement of the message interval because an increasing message interval inevitably creates a large number of redundant messages in the entire network. However, in our simulation, the buffer resource is always set to be in a relatively constrained range so that the incurred frequent message drop would introduce additional hop counts into the routing process. Based on the above analysis, we can conclude that GINS provides better routing performance compared with GRONE and LPHU.

6 Conclusion

In this paper, we propose an algorithm for DTNs called GINS. We do not assume the position of the destination node. Thus, a good way to improve the delivery probability and lower the average hop count is to spread the message uniformly. We therefore define the utility function and the forwarding willingness degree to select the relay node. The utility function takes transmission direction and node distance into consideration, thereby ensuring that the message is distributed in a radiating manner. We also present a more effective message and buffer management method. Simulation results show that GINS achieves a better performance than GRONE and LPHU in terms of message delivery ratio, overhead ratio, and average hop count. In the future, we will focus on a more precise evaluation criterion for relay node selection, which can better adapt to specific application scenarios.

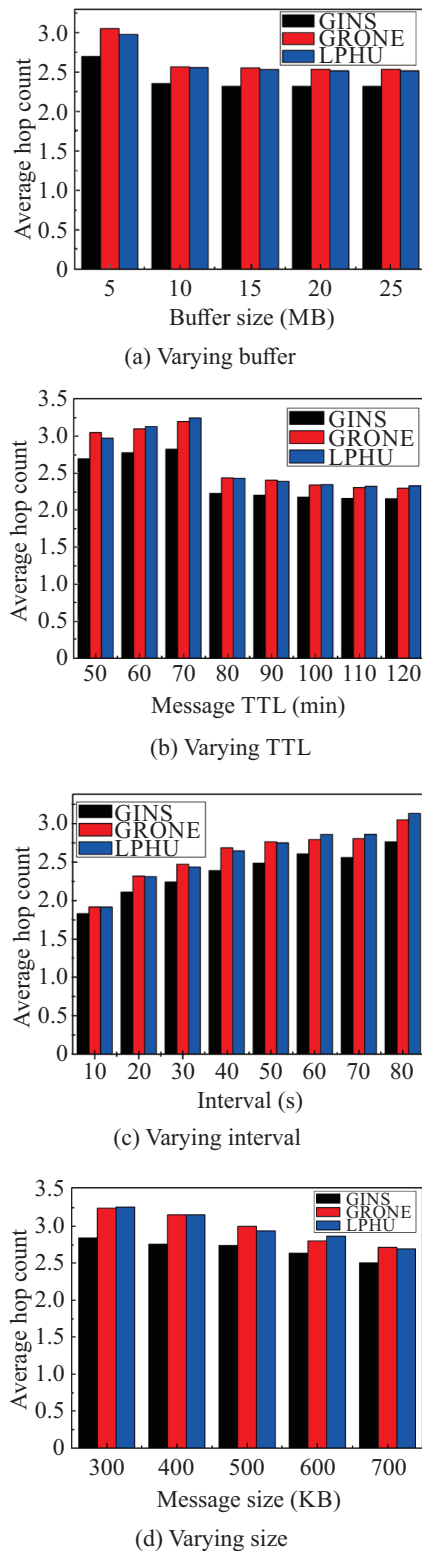


Fig. 6 Average hop count versus buffer size, message TTL, message interval, and message size.

Acknowledgment

This research was supported in part by the National Natural Science Foundation of China (Nos. 61502261,

61572457, and 61379132), and the Science and Technology Plan Project for Colleges and Universities of Shandong Province (No. J14LN85).

References

- [1] B. Soelistijanto and M. P. Howarth, Transfer reliability and congestion control strategies in opportunistic networks: A survey, *IEEE Communications Surveys Tutorials*, vol. 64, no. 1, pp. 538–555, 2014.
- [2] S. Cha, E. Talipov, and H. Cha, Data delivery scheme for intermittently connected mobile sensor networks, *Computer Communications*, vol. 36, no. 5, pp. 504–519, 2013.
- [3] Z. Ren, S. Peng, H. Chen, H. Fang, and Q. Chen, Epidemic routing based on adaptive compression of vectors: Efficient low-delay routing for opportunistic networks based on adaptive compression of vectors, *International Journal of Communication Systems*, vol. 28, no. 3, pp. 560–573, 2015.
- [4] Y. Cao and Z. Sun, Routing in delay/disruption tolerant networks: A taxonomy, survey and challenges, *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, p. 654, 2013.
- [5] A. Picu and T. Spyropoulos, Distributed stochastic optimization in opportunistic networks: The case of optimal relay selection, in *Proc. 5th ACM Workshop Challenged Netw.*, Chicago, IL, USA, Sep. 20–24, 2010, p. 2128.
- [6] Z. Mao, Y. Jiang, G. Min, S. Leng, X. Jin, and K. Yang, Mobile social networks: Design requirement, architecture, and state-of-art technology, *Computer Communications*, vol. 100, pp. 1–19, 2017.
- [7] M. Xiao, J. Wu, and L. Huang, Community-aware opportunistic routing in mobile social networks, *IEEE Transactions on Computers*, vol. 63, no. 7, pp. 1682–1695, 2014.
- [8] M. Motani, V. Srinivasan, and P. S. Nuggehalli, Peoplenet: Engineering a wireless virtual social network, in *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking, MobiCom'05*, New York, NY, USA, 2005, pp. 243–257.
- [9] P. Sermpezis and T. Spyropoulos, Understanding the effects of social selfishness on the performance of heterogeneous opportunistic networks, *Computer Communications*, vol. 48, pp. 71–83, 2014.
- [10] R.-I. Ciobanu, C. Dober, M. Dascalu, Ș. Trăușan-Matu, and V. Cristea, SENSE: A collaborative selfish node detection and incentive mechanism for opportunistic networks, *Journal of Network and Computer Applications*, vol. 41, pp. 240–249, 2014.
- [11] D. Pan, H. Zhang, and W. Chen, Transmission of multimedia contents in opportunistic networks with social selfish nodes, *Multimedia Systems*, vol. 21, no. 3, pp. 277–288, 2015.
- [12] L. Boloni and D. Turgut, Protecting bridge: Reorganizing sensor networks after catastrophic events, in *2011 7th International Wireless Communications and Mobile Computing Conference*, IEEE, 2011, pp. 2028–2033.

- [13] J. Xu, J. Li, L. You, and C. Dai, Lphu: A local position and history utility based routing algorithm for delay tolerant networks, *Sensors & Transducers*, vol. 157, no. 10, p. 419, 2013.
- [14] L. You, J. Li, C. Wei, and C. Dai, A one-hop information based geographic routing protocol for delay tolerant manets, *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 20, no. 2, pp. 107–122, 2015.
- [15] A. Balasubramanian, B. N. Levine, and A. Venkataramani, Replication routing in DTNs: A resource allocation approach, *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, pp. 596–609, 2010.
- [16] A. Martin-Campilo, J. Crowcroft, E. Yoneki, and R. Marti, Evaluating opportunistic networks in disaster scenarios, *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 870–880, 2013.
- [17] Q. Li, S. Zhu, and G. Cao, Routing in socially selfish delay tolerant networks, in *IEEE INFOCOM*, 2010.
- [18] S. C. Nelson, M. Bakht, and R. Kravets, Encounter-based routing in DTNs, in *Proceedings of the 28th Conference on Computer Communications (IEEE INFOCOM'09)*, 2009, pp. 846–854.



Fang Lu is currently pursuing the master degree at the School of Computer Science and Technology, Qingdao University. She received the bachelor degree in computer science and technology from the University of Computer Science and Technology of Linyi, China in 2015. Her research interest is delay/disruption tolerant networks and mobile social networks.



Youmei Song is currently pursuing the master degree at the School of Computer Science and Technology, Qingdao University. She received the bachelor degree in computer science and technology from Shenyang Ligong University, China in 2014. Her research interests include delay/disruption tolerant networks.



Jianbo Li is currently a professor at the Computer Science and Technology College of Qingdao University. He received the bachelor and master degrees in computer science from Qingdao University, China in 2002 and 2005, respectively, and the PhD degree in computer science from the University of Science and Technology of China in 2009. His research interests include delay/disruption tolerant networks, mobile social networks, and data offloading.



Fushu Wang is currently pursuing the master degree at the School of Computer Science and Technology, Qingdao University. He received the bachelor degree from Qingdao University, China in 2015. His research interests include delay/disruption tolerant networks.



Shan Jiang is currently a lecturer at Qingdao University. She received the bachelor and master degrees in information management from Shandong Institute of Business and Technology in 2004 and Nankai University in 2007. Her research interests include delay/disruption tolerant networks and algorithm design and analysis.

analysis.